UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,490	11/14/2003	Zaki Chasmawala	5150-76801	1049

7590 08/20/2008

Jeffrey C. Hood
Meyertons, Hood, Kivlin, Kowert & Goetzel
P.O. Box 398
Austin, TX 78767

EXAMINER	
CEHIC, KENAN	

ART UNIT	PAPER NUMBER
2616	

MAIL DATE	DELIVERY MODE
08/20/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/714,490	CHASMAWALA ET AL.	
	Examiner	Art Unit	
	KENAN CEHIC	2616	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05/07/2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *63-90* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *63-90* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

1. Claims 81-90 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

For claim 81, the limitations "arrangement of arbitration IDs for the plurality channels in the first CAN message" is not supported by the original disclosure. The above language implies that the method uses *multiple* arbitration IDs for a single to define one specific configuration/set of channel inside *one* CAN message i.e. arbitration IDs X, Y, Z are associated with one specific set of channels (for example channel I, II) that are in *a CAN message*. However, the specification (especially page 9-10), describes that *one specific* arbitration ID is used to associate a specific set of channels in a CAN message i.e. associate arbitration ID X with channel I, II for a CAN message.

For claim 83, the same reasoning applies as to claim 81 for the recited limitations "CAN message comprises one or more message arbitration IDs, wherein each one or more message arbitration IDs identifies the type and location of data in the first CAN message".

Dependent claims are rejected since they depend on a rejected claim.

2. Claims 82, 83 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

For claim 82, "the first configuration" has no antecedent basis.

Dependent claims are rejected since they depend on a rejected claim

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

3. Claim 63, 66-68, 70, 71, 73, 76-78, 80 are rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") in view of "Datasheet CANopen Device Monitor", hereinafter D1.

For claim 63, Boterenbrood discloses querying (see page 11-12 3.6.5 “Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload” and see page 3 left column “DBT master…master” and page 5 both columns “SDO provides a client access to entries…of a device OD” and page 11 right column “SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO” and page 12 left column “Upload means reading from the object dictionary.”) a first network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) for configuration of a first data message (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO”), wherein the first network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) is coupled to a network network (see fig 1 and page 2 both column “CAN networks”), wherein the first network device is configured to send (see page 5 3. “PDO…used to transfer real-time data….producer to one or more consumers.” and fig 1)coupled to the CAN network (see fig 1 and page 2 both column “CAN networks”) the first data message (see page 5 “Process Data Object…transfer real-time data…digital I/O

values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") to a second network device coupled to the network (see page 5 3. "PDO…used to transfer real-time data….producer to one or more consumers." and fig 1) coupled to the CAN network (see fig 1 and page 2 both column "CAN networks");

sending a first configuration message (see page 5 left and right column "Service Data Object" and page 11 right column "SDO") to a configuration computer (see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary.") in response to said querying (see page 11-12 3.6.5 "Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload" and see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary."), wherein the first configuration message (see page 3 left column "DBT master…master" and page 5 both

columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary."; page 5 left and right column "Service Data Object" and page 11 right column "SDO") comprises a first configuration for a plurality of data elements in the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO");

overriding the first configuration of the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible"), wherein said overriding comprises selecting a second configuration of the first data message (see page 5 "Process Data Object…transfer real-

time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible"); and

sending a second configuration message (see page 9 right column "COB-ID allocation via SDO" and pages 11-12 l "SDO message…is written to Object Dictionary.." and page 8 "SDO to write new values") to the first network device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer"), wherein the second configuration message (see page 9 right column "COB-ID allocation via SDO" and pages 11-12 l "SDO message…is written to Object Dictionary.." and page 8 "SDO to write new values") comprises the second configuration for the plurality of data elements in the first data message (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO" and page 5

"Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO");

wherein the first network device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") is configured to send the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO") according to the second configuration (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO") in response to receiving the second configuration message (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column

"COB-ID allocation via SDO is possible") ; configuration of the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible")

For claim 66, Boterenbrood discloses wherein the first network device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") comprises a first input module and a second input module (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input"), wherein the first input module is configured to acquire first physical data, (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input") wherein the second input module is configured to acquire second physical data (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input"); and

wherein the first data message contains data for the first physical data and the second physical data (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and

page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO” and see page 5 right column “4 16-bit analogue inputs” and page 11 3.64. “I/O modules…digital input..analog input”).

For claim 67, Boterenbrood discloses wherein the first network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) comprises at least one of two or more inputs (see page 5 right column “4 16-bit analogue inputs” and page 11 3.64. “I/O modules…digital input..analog input”); and

wherein the first data message (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO”) contains data for the two or more inputs (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO” and see page 5 right column “4 16-bit analogue inputs” and page 11 3.64. “I/O modules…digital input..analog input”)

For claim 70, Boterenbrood discloses wherein the first network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is

the server" and page 11 right column “server” and page 5 3. “PDO…producer”) is further configured to said send the first data

message (see page 3 left column “DBT master…master” and page 5 both columns “SDO provides a client access to entries…of a device OD” and page 11 right column “SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command

Specifier…upload…request/response….request/response…in SDO” and page 12 left column “Upload means reading from the object dictionary.”) on occurrence of one or more events, wherein the one or more events comprise: poll from the communication network (see page 11-12 3.6.5 “Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload”).

For claim 71, Boterenbrood A network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) for use in a network (see fig 1 and page 2 both column “CAN networks”), wherein the network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) comprises:

a network module (see page 1, fig 1, OSI layer 1 physical layer and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) coupled to a network (see fig 1 and page 2 both column “CAN networks”); a first input module (see fig 1; Data link

layer, OSI layer 7 application layer) coupled to the network module (see fig 1); wherein the network module (see page 1, fig 1, OSI layer 1 physical layer) is operable to transmit a first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") on the network (see page 5 3. "PDO…used to transfer real-time data….producer to one or more consumers." and fig 1)coupled to the CAN network (see fig 1 and page 2 both column "CAN networks"), wherein the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") comprises a plurality of data elements ordered according to a first configuration (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO");

wherein the first configuration of the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the

contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") is operable to be re- configured into a second configuration (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible"), wherein the computer (see page 5 3.2 "SDO provides a client access to entries…of a device OD") is coupled to the network (see fig 1 and page 2 both column "CAN networks"), wherein the first data message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") is operable to be re-configured according to the second configuration (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping

Parameter...objects...mapped into the PDO" and see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible").

For claim 73, Boterenbrood discloses wherein the computer is operable to send a configuration message (see page 9 right column "COB-ID allocation via SDO" and pages 11-12 l "SDO message…is written to Object Dictionary.." and page 8 "SDO to write new values") to the network module (see page 1, fig 1, OSI layer 1 physical layer and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") ; and wherein said re-configuring the first data message comprises re-configuring the first data message (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO" and page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO") in response to receiving the configuration message (see page 9 right column

"COB-ID allocation via SDO" and pages 11-12 l "SDO message…is written to Object Dictionary.." and page 8 "SDO to write new values").

For claim 76, Boterenbrood discloses wherein the first network device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") comprises a first input module and a second input module (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input"), wherein the first input module is configured to acquire first physical data, (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input") wherein the second input module is configured to acquire second physical data (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input"); and

wherein the first data message contains data for the first physical data and the second physical data (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input")

For claim 77, Boterenbrood discloses wherein the first network device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer")

comprises at least one of two or more inputs (see page 5 right column “4 16-bit analogue inputs” and page 11 3.64. “I/O modules…digital input..analog input”); and

wherein the first data message (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO”) contains data for the two or more inputs (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO” and see page 5 right column “4 16-bit analogue inputs” and page 11 3.64. “I/O modules…digital input..analog input”)

For claim 80, Boterenbrood discloses wherein the first network device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”) is further configured to said send the first data

message (see page 3 left column “DBT master…master” and page 5 both columns “SDO provides a client access to entries…of a device OD” and page 11 right column “SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command

Specifier…upload…request/response….request/response…in SDO” and page 12 left

column "Upload means reading from the object dictionary.") on occurrence of one or more events, wherein the one or more events comprise: poll from the communication network (see page 11-12 3.6.5 "Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload").

Boterenbrood is silent about:

For claim 63, displaying the first configuration on a display of the configuration computer; and user selecting the configuration;

For claim 68 and 78, wherein the configuration computer is configured to execute a graphical program, wherein the graphical program is configured to communicate with the first network device; and wherein the graphical program is operable to process and use data from the first data message.

For claim 71, using a graphical configuration tool, wherein the graphical configuration tool executes on a computer.

D1 from the same field of endeavor discloses software with the following features:

For claim 63, D1 discloses displaying the first configuration on a display of the configuration computer (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…" and under System Requirements "PC's wit Microsoft Windows 9X/NT ...systems..."); user selecting the configuration (see under Description "reading and writing of device parameters with CANopen SDO transfer messages…writing of a device object…commands for configuring PDOs")

For claim 68, and 78, D1 discloses wherein the configuration computer is configured to execute a graphical program, wherein the graphical program (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…" and under System Requirements "PC's wit Microsoft Windows 9X/NT ...systems...") is configured to communicate with the first network device (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…transmitting or receiving PDOs"); and wherein the graphical program is operable to process and use data from the first data message (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…").

For claim 71, D1 discloses using a graphical configuration tool (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…" and under System Requirements "PC's wit Microsoft Windows 9X/NT ...systems..."), wherein the graphical configuration tool executes on a computer (see under Application "Displaying all implemented device parameters and data…description of the object

dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…" and under System Requirements "PC's wit Microsoft Windows 9X/NT ...systems...")

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood by using the features, as taught by D1, in order to provide a device monitor tool for CANopen devices / networks which makes it possible to access CANopen services with minimum effort and provides easy inspection and configuration of CANopen devices (see D1 page 1)

4. Claim 64, 74 rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") and "Datasheet CANopen Device Monitor", hereinafter D1 as applied to claim 63/73 above, further in view of Farsi et al ("CANopen: CONFIGURE and DEVICE TESTING")

For claim 64, and 74 Boterenbrood and D1 discloses the claimed invention as described in paragraph 3.

Boterenbrood and D1 are silent about:

For claim 64, and 74 wherein the first configuration comprises location and type of each data element of the plurality of data elements; and

wherein said user selecting comprises the user selecting and rearranging on the display the location and type of one or more data elements of the plurality of data elements of the first configuration.

Farsi from the same or similar field of endeavor discloses a communication network with the following features:

For claim 64, and 74 Farsi discloses wherein the first configuration comprises location and type of each data element of the plurality of data channels (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5); wherein said user selecting comprises the user selecting and rearranging (see page 378 4. "user full control …full access to network management functions…In network configuration mode the user can select and assign object dictionaries to nodes…new object dictionaries...configuration of process data objects (PDO maps)….Allocation of identifiers for communication…process data objects…configured process data maps…user can easily opt to download or upload service data from any of the devices…selecting the required objects from corresponding object dictionary" and page 376- 377 "PDO Mapping…wish to configure…input operational telegram to have the following structure….write the following value to the corresponding PDO-Mapping structure…" and Table 3-4) on the display (see page 377-378 "VGA graphics display…Graphical user interface") the location and type of one or more data elements of the plurality of data elements of the first configuration (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5).

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood and D1 by using the features, as taught by Farsi, in order to provide a configuration and testing tool for a CANopen network

/devices that is inexpensive way of learning CANopen functionality and can be used to isolate faults and malfunctions of individual components (see Farsi page 373)

5. Claim 65, and 75 rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus"), "Datasheet CANopen Device Monitor"(D1), and of Farsi et al ("CANopen: CONFIGURE and DEVICE TESTING") as applied to claim 64 / 74 above, further in view of "CAN Technical introduction", hereinafter D2.

For claim 65, and 75 Boterenbrood, D1, and Farsi discloses the claimed invention as described in paragraph 4.

For claim 65, and 75 Boterenbrood discloses CAN IDs identifies the type of data in the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO")

For claim 65, and 75 Farsi further discloses identifies the location (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5).

Boterenbrood, D1, and Farsi are silent about:

For claim 65, and 75 CAN Ids are used for arbitration

D2 from the same or similar field of endeavor discloses a communication network with the following features:

For claim 65, and 75 D2 discloses CAN Ids are used for arbitration (see page 3-4 under headings Non-destructive bitwise arbitration and Efficiency of bus allocation, "Bus access conflicts are resolved by bitwise arbitration on the identifiers…..The method of bitwise arbitration using the identifier of the message to be transmitted uniquely resolves any collision…" and page 6 figure titled "Message frame for standart format", "Arbitration Field…11 bit identifier" and page 6 under heading Message frame formats "arbitration field..identifier")

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood, D1, and Farsi by using the features, as taught by D2, in order to provide non-destructive bitwise arbitration on a CAN network, where data is not lost during arbitration and provides an increased efficiency of the bus arbitration process (see D2 page 3, and 5).

6. Claim 69 and 79 is rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") in view of "Datasheet CANopen Device Monitor",(D1) as applied to claim 68 / 78 above, further in view of Kodosky et al. (5,475,851).

For claim 69 and 79 Boterenbrood and D1 discloses the claimed invention as described in paragraph 3.

Boterenbrood and D1 are silent about:

For claim 69 and 79 ,wherein the graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the graphical program; and wherein the graphical program comprises a block diagram portion and a user interface portion.

Kodosky from the same or similar field of endeavor discloses a communication network with the following features:

For claim 69 and 79, Kodosky discloses wherein the graphical program comprises a plurality of interconnected nodes (see Figure 22 and 43; note the interconnected nodes) that visually indicate functionality of the graphical program (see Figure 22 and 43; the nodes indicate the functionality of the program); and

wherein the graphical program comprises a block diagram portion (see Figure 3, Figure; Block diagram) and a user interface portion (see Figure 124).

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood and D1 by using the features, as taught by Kodosky, in order that the user is able to read and adjust the values of variables during program execution (see column 6 lines 11-16 of Kodosky et al.) and to visual the program flow.

7. Claim 72 are rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") and "Datasheet CANopen Device Monitor" (D1) as applied to claim 71 above, further in view of Albert et al (US 2006/0107746).

For claim 72, Boterenbrood and D1 discloses the claimed invention as described in paragraph 3.

For claim 72, Boterenbrood further discloses wherein the first input module (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input") is operable to acquire first physical data (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input"), wherein the plurality of data elements comprises the first digital data (see page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input"); and wherein the first data message is configured to contain the first digital data (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and page 5 right column "4 16-bit analogue inputs" and page 11 3.64. "I/O modules…digital input..analog input").

Boterenbrood and D1 silent about:

For claim 72, wherein the first physical data is operable to be digitized into first digital data

Albert from the same or similar field of endeavor discloses a communication network with the following features:

For claim 72, Albert discloses wherein the first physical data is operable to be digitized into first digital data (see section 0012 "digitized input from a sensor analog front" and

section 0021 "digitized…analog-to-digital converter" and section 0057 "digitized signal received from the analog-to-digital converter")

The prior art as disclosed in Albert discloses the claimed features of converting analog data to a digital format. A person of ordinary skill in the art could have combined this features into the system of Boterenbrood and D1 and the elements would have performed merely their function as they did separately. One of ordinary skill in the art would have recognized that the results of combining this feature would have yielded predictable results.

8. Claim 81-85 rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") in view of Farsi et al ("CANopen: CONFIGURE and DEVICE TESTING"), "Datasheet CANopen Device Monitor", hereinafter D1 and "CAN Technical introduction", hereinafter D2.

For claim 81, Boterenbrood discloses a first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") sending a first configuration message (see page 5 left and right column "Service Data Object" and page 11 right column "SDO") to a client/master (see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary."), wherein the first configuration

message (see page 5 left and right column "Service Data Object" and page 11 right column "SDO") to a client/master (see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary.") comprises a first arrangement of CAN IDs (see page 3 left column "COB-ID…CAN standard frame…11-bit identifier…29-bit identifier" and page 5 right column "SDO…" and Table 8 "COB-ID used by PDO" ; identifiers used for PDOs are dictionary entries;) for a plurality of data channels in a first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO"), wherein the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") is operable to be sent from the first CAN device to a second CAN device (see page 5 3. "PDO…used to transfer real-time data….producer to one or more consumers." and fig 1), wherein both the first and the second CAN devices (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page

5 3. "PDO…producer…consumer") are coupled to a CAN network (see fig 1 and page 2 both column "CAN networks");

changing the first arrangement of CAN IDs (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible") for the plurality of data channels in the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO"), wherein said changing comprises the selecting a second arrangement of CAN IDs (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO")) for the plurality of data channels in the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO");

sending a second configuration message (see page 9 right column "COB-ID allocation via SDO" and pages 11-12 l "SDO message…is written to Object Dictionary.." and page 8 "SDO to write new values") to the first CAN device (see page 9 right column "CANopen devices"), wherein the second configuration message comprises the second arrangement of CAN IDs message (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO")) for the plurality of data channels in the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO"); and

the first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") sending the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO")in accordance to the second arrangement of CAN IDs (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO

identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO")) for the plurality of data channels (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO") in response to receiving the second configuration message (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible"); the arrangement of CAN IDs for the plurality of data channels in the first can message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO") being object dictionary entries/parameters (see page 3 left column "COB-ID…CAN standard frame…11-bit identifier…29-bit identifier" and page 5 right column "SDO…" and Table 8 "COB-ID used by PDO" ; identifiers used for PDOs are dictionary entries;).

For claim 83, Boterenbrood discloses wherein the first CAN message comprises one or more

message CAN IDs (see page 3 left column "COB-ID…CAN standard frame…11-bit identifier…29-bit identifier" and page 5 right column "SDO…" and Table 8 "COB-ID used by PDO" ; identifiers used for PDOs are dictionary entries;), wherein each one of the one or more message CAN IDs identifies the type of data in the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO").

For claim 85, Boterenbrood discloses wherein the first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") is further configured to said send the first CAN message (see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary.") on occurrence of one or more events, wherein the one or more events comprise: or poll from the communication network (see page 11-12 3.6.5 "Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload").

Boterenbrood is silent about:

For claim 81,CAN ID is used for arbitration, the user selecting the configuration, the host computer, and displaying of the dictionary entires/parameters.

For claim 82, wherein the first configuration comprises location and type of each data element of the plurality of data channels; and

wherein said user selecting comprises the user selecting and rearranging on the display the location and type of one or more data channels of the plurality of data channels of the first configuration.

For claim 83, the location of the data and CAN ID is used for arbitration

For claim 84, wherein the host computer is configured to execute a graphical program, wherein the graphical program is configured to communicate with the first CAN device; and wherein the graphical program is operable to process and use data from the first CAN message.

Farsi from the same field of endeavor discloses a configuration tool/program for a CANopen network with the following features:

For claim 81, Farsi discloses, , the host computer (see page 377 right column "IBM-compatible PC….CDT software"); the user selecting the configuration (see page 378 4. "user full control …full access to network management functions…In network configuration mode the user can select and assign object dictionaries to nodes…new object dictionaries...configuration of process data objects (PDO maps)….Allocation of identifiers for communication…process data objects…configured process data maps…user can easily opt to download or upload service data from any of the devices…selecting the required objects from corresponding object dictionary" and page 376- 377 "PDO Mapping…wish to configure…input operational telegram to have the following structure….write the following value to the corresponding PDO-Mapping structure…" and Table 3-5).

For claim 82, Farsi discloses wherein the first configuration comprises location and type of each data element of the plurality of data channels (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5); and

wherein said user selecting comprises the user selecting and rearranging on the location and type of one or more data channels (see page 378 4. "user full control …full access to network management fuctions…In network configuration mode the user can select and assign object dictionaries to nodes…new object dictionaries...configuration of process data objects (PDO maps)….Allocation of identifiers for communication…process data objects…configured process data maps…user can easily opt to download or upload service data from any of the devices…selecting the required objects from corresponding object dictionary" and page 376- 377 "PDO Mapping…wish to configure…input operational telegram to have the following structure….write the following value to the corresponding PDO-Mapping structure…" and Table 3-5) of the plurality of data channels of the first configuration (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5).

For claim 83, Farsi discloses the location of the data (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5) and CAN ID is used for arbitration (see page 375 left column "guaranteed bus priority levels…identifiers…devices using high priority identifier…messages identifiers such that all PDO message receive a higher priority…")

D1 from the same field of endeavor discloses a computer program with the following features:

For claim 81, D1 discloses displaying the dictionary entries/parameters (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…"); user selecting the configuration (see under Description "reading and writing of device parameters with CANopen SDO transfer messages…writing of a device object…commands for configuring PDOs")

For claim 82, D1 discloses the selecting (see under Description "reading and writing of device parameters with CANopen SDO transfer messages…writing of a device object…commands for configuring PDOs") on the display (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…)

For claim 84, D1 discloses wherein the host computer is configured to execute a graphical program (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…" and under System Requirements "PC's wit Microsoft Windows 9X/NT ...systems..."), wherein the graphical program is configured to communicate with the first CAN device (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…transmitting or receiving PDOs"); and wherein the graphical program is

operable to process and use data from the first CAN message (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…").

D2 from the same field of endeavor discloses:

For claim 81, D2 discloses CAN ID is used for arbitration (see page 3-4 under headings Non-destructive bitwise arbitration and Efficiency of bus allocation, "Bus access conflicts are resolved by bitwise arbitration on the identifiers…..The method of bitwise arbitration using the identifier of the message to be transmitted uniquely resolves any collision…" and page 6 figure titled "Message frame for standart format", "Arbitration Field…11 bit identifier" and page 6 under heading Message frame formats "arbitration field..identifier")

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood by using the features, as taught by D1, Farsi, and D2,; in order to provide a device monitor tool for CANopen devices / networks which makes it possible to access CANopen services with minimum effort and provides easy inspection and configuration of CANopen devices (see D1 page 1); in order to provide a configuration and testing tool for a CANopen network /devices that is inexpensive way of learning CANopen functionality and can be used to isolate faults and malfunctions of individual components (see Farsi page 373); in order to provide non-destructive bitwise arbitration on a CAN network, where data is not lost during arbitration and provides an increased efficiency of the bus arbitration process (see D2 page 3, and 5).

9. Claim 86, 89, 90 rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") in view of "Datasheet CANopen Device Monitor", hereinafter D1.

For claim 86, Boterenbrood discloses querying (see page 11-12 3.6.5 "Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload" and see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary.") a first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") for configuration of a first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO"), wherein the first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") is coupled to a CAN network (see fig 1 and page 2 both column "CAN networks"), wherein the first CAN message (see page 5 "Process Data

Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter…objects...mapped into the PDO") is operable to be sent from the first CAN device to a second CAN device (see page 5 3. "PDO…used to transfer real-time data….producer to one or more consumers." and fig 1)coupled to the CAN network (see fig 1 and page 2 both column "CAN networks");

sending a first configuration message (see page 5 left and right column "Service Data Object" and page 11 right column "SDO") to a client/master (see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary.") to a host computer in response to said querying (see page 11-12 3.6.5 "Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload" and see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary."), wherein the first

configuration message (see page 3 left column “DBT master…master” and page 5 both columns “SDO provides a client access to entries…of a device OD” and page 11 right column “SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO” and page 12 left column “Upload means reading from the object dictionary.”; page 5 left and right column "Service Data Object" and page 11 right column “SDO”) comprises a first configuration for a plurality of data channels in the first CAN message (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO”);

changing the first configuration of the first CAN message (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO” and see page 8, 3.4 “CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated” and page 9 right column “COB-ID allocation via SDO is possible”), wherein said changing comprises the selecting a second

configuration of the first CAN message (see page 5 “Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only” and page 11 3.6.4 “PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping” and page 5 right column “PDO Mapping Parameter...objects...mapped into the PDO” and see page 8, 3.4 “CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated” and page 9 right column “COB-ID allocation via SDO is possible”);

sending a second configuration message (see page 9 right column “COB-ID allocation via SDO” and pages 11-12 l “SDO message…is written to Object Dictionary..” and page 8 “SDO to write new values”) to the first CAN device (see page 1, fig 1 and see page 3, left column “slave” and page 5 left and right column "device….device is the server" and page 11 right column “server” and page 5 3. “PDO…producer”), wherein the second configuration message (see page 9 right column “COB-ID allocation via SDO” and pages 11-12 l “SDO message…is written to Object Dictionary..” and page 8 “SDO to write new values”) comprises the second configuration for the plurality of data channels in the first CAN message (see page 8, 3.4 “CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated” and page 9 right column “COB-ID allocation via SDO is possible” and page 6 left column “PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping” and page 5

right column "PDO Mapping Parameter..objects...mapped into the PDO" and page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO"); and the first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") sending the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO") to the second CAN device (see page 5 3. "PDO…used to transfer real-time data….producer to one or more consumers." and fig 1), wherein the first CAN message is created in accordance with the second configuration (see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible" and page 6 left column "PDO Mapping parameter...is configurable using SDO messages...variable PDO mapping" and page 5 right column "PDO Mapping Parameter..objects...mapped into the PDO"); the first configuration of the first CAN message (see page 5 "Process Data Object…transfer real-

time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO"); and second configuration of the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO" and see page 8, 3.4 "CANopen identifier distribution…PDO mapping…modifying the PDO identifiers…using…SDO…CAL DBT…services…CAN identifiers for communication of SDOs and PDOs are allocated" and page 9 right column "COB-ID allocation via SDO is possible")

For claim 90, Boterenbrood discloses wherein the first CAN device (see page 1, fig 1 and see page 3, left column "slave" and page 5 left and right column "device….device is the server" and page 11 right column "server" and page 5 3. "PDO…producer") is further configured to said send the first CAN message (see page 3 left column "DBT master…master" and page 5 both columns "SDO provides a client access to entries…of a device OD" and page 11 right column "SDO…used to access the Object Dictionary of a device…requester…client…Client CAN-message…SDO command Specifier…upload…request/response….request/response…in SDO" and page 12 left column "Upload means reading from the object dictionary.") on occurrence of one or

more events, wherein the one or more events comprise: or poll from the communication network (see page 11-12 3.6.5 "Client CAN-message…Client Request..SDO Command Specifier….upload…Request…upload").

Boterenbrood is silent about:

For claim 86, displaying the first configuration on a display of the host computer; changing the displayed first configuration; user selecting a second configuration

For claim 89, wherein the host computer is configured to execute a graphical program, wherein the graphical program is configured to communicate with the first CAN device; and wherein the graphical program is operable to process and use data from the first CAN message.

D1 from the same field of endeavor discloses software with the following features:

For claim 86, D1 discloses displaying the first configuration on a display of the host computer (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…" and under System Requirements "PC's wit Microsoft Windows 9X/NT ...systems..."); changing (see under Description "reading and writing of device parameters with CANopen SDO transfer messages…writing of a device object…commands for configuring PDOs") the displayed first configuration (see under Application "Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…"; under Description "Graphical user interface..representation of the object dictionary…"

and under System Requirements “PC’s wit Microsoft Windows 9X/NT ...systems..."); user selecting a second configuration see under Description ”reading and writing of device parameters with CANopen SDO transfer messages…writing of a device object…commands for configuring PDOs”).

For claim 89, D1 discloses wherein the host computer is configured to execute a graphical program (see under Application “Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…”; under Description “Graphical user interface..representation of the object dictionary…” and under System Requirements “PC’s wit Microsoft Windows 9X/NT ...systems..."), wherein the graphical program is configured to communicate with the first CAN device (see under Application “Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…”; under Description “Graphical user interface..representation of the object dictionary…transmitting or receiving PDOs”); and wherein the graphical program is operable to process and use data from the first CAN message (see under Application “Displaying all implemented device parameters and data…description of the object dictionary entries is concurrently available…graphical representation of value…”; under Description “Graphical user interface..representation of the object dictionary…”).

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood by using the features, as taught by D1, in order to provide a device monitor tool for CANopen devices / networks which makes it

possible to access CANopen services with minimum effort and provides easy inspection and configuration of CANopen devices (see D1 page 1)

10. Claim 87 rejected under 35 U.S.C. 103(a) as being unpatentable in view of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") and "Datasheet CANopen Device Monitor" (D1) as applied to claim 86 above, further in view of Farsi et al ("CANopen: CONFIGURE and DEVICE TESTING").

For claim 87, Boterenbrood and D1 discloses the claimed invention as described in paragraph 9.

Boterenbrood and D1 are silent about:

For claim 87, wherein the first configuration comprises location and type of each data element of the plurality of data channels; and wherein said user selecting comprises selecting and rearranging on the display the location and type of one or more data channels of the plurality of data channels of the first configuration.

Farsi from the same or similar field of endeavor discloses a communication network with the following features:

For claim 87, Farsi discloses wherein the first configuration comprises location and type of each data element of the plurality of data channels (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5); and

wherein said user selecting comprises selecting and rearranging (see page 378 4. "user full control …full access to network management functions…In network configuration

mode the user can select and assign object dictionaries to nodes…new object dictionaries...configuration of process data objects (PDO maps)….Allocation of identifiers for communication…process data objects…configured process data maps…user can easily opt to download or upload service data from any of the devices…selecting the required objects from corresponding object dictionary" and page 376- 377 "PDO Mapping…wish to configure…input operational telegram to have the following structure….write the following value to the corresponding PDO-Mapping structure…" and Table 3-4) on the display (see page 377-378 "VGA graphics display…Graphical user interface") the location and type of one or more data channels of the plurality of data channels of the first configuration (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5).

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood and D1 by using the features, as taught by Farsi in order to provide a configuration and testing tool for a CANopen network /devices that is inexpensive way of learning CANopen functionality and can be used to isolate faults and malfunctions of individual components (see Farsi page 373);

11. Claim 88 is rejected under 35 U.S.C. 103(a) as being unpatentable in view of of H. Boterenbrood ("CANopen high-level protocol for CAN-bus") and "Datasheet CANopen Device Monitor" (D1) and Farsi et al ("CANopen: CONFIGURE and DEVICE TESTING") as applied above to claim 87, further in view of "CAN Technical introduction", hereinafter D2.

For claim 88, Boterenbrood, D1, and Farsi discloses the claimed invention as described in paragraph 10.

For claim 88, Boterenbrood disclose CAN ID identifies (see page 3 left column "COB-ID…CAN standard frame…11-bit identifier…29-bit identifier" and page 5 right column "SDO…" and Table 8 "COB-ID used by PDO" ; identifiers used for PDOs are dictionary entries;) a type of data in the first CAN message (see page 5 "Process Data Object…transfer real-time data…digital I/O values…inputs…data content of a PDO is defined through its CAN-identifier only" and page 11 3.6.4 "PDO…changing the contents of Object 0x1A01 the content of the PDO can be change…variable PDO mapping" and page 5 right column "PDO Mapping Parameter...objects...mapped into the PDO")

For claim 88, Farsi discloses identifies the location of data in the first CAN message (see Farsi pages 376-377, 2.3.3 "PDO Mapping…following structure….PDO mapping Structure…following structure…PDO mapping structure", Tables 3-5).

Boterenbrood, D1, and Farsi are silent about:

For claim 88, wherein the first CAN message comprises one or more message arbitration IDs, wherein each one of the one or more message arbitration Ids; and that a CAN ID is a used for arbitration.

D2 from the same or similar field of endeavor discloses a communication network with the following features:

For claim 88, D2 discloses, wherein the first CAN message comprises one or more message arbitration Ids (see page 3-4 under headings Non-destructive bitwise arbitration

and Efficiency of bus allocation, “Bus access conflicts are resolved by bitwise arbitration on the identifiers…..The method of bitwise arbitration using the identifier of the message to be transmitted uniquely resolves any collision…” and page 6 figure titled “Message frame for standart format”, “Arbitration Field…11 bit identifier” and page 6 under heading Message frame formats "arbitration field..identifier")

, wherein each one of the one or more message arbitration Ids (see page 3-4 under headings Non-destructive bitwise arbitration and Efficiency of bus allocation, “Bus access conflicts are resolved by bitwise arbitration on the identifiers…..The method of bitwise arbitration using the identifier of the message to be transmitted uniquely resolves any collision…” and page 6 figure titled “Message frame for standart format”, “Arbitration Field…11 bit identifier” and page 6 under heading Message frame formats "arbitration field..identifier"); and that a CAN ID is a used for arbitration (see page 3-4 under headings Non-destructive bitwise arbitration and Efficiency of bus allocation, “Bus access conflicts are resolved by bitwise arbitration on the identifiers…..The method of bitwise arbitration using the identifier of the message to be transmitted uniquely resolves any collision…” and page 6 figure titled “Message frame for standart format”, “Arbitration Field…11 bit identifier” and page 6 under heading Message frame formats "arbitration field..identifier").

It would have been obvious to one of the ordinary skill in the art at the time of the invention to modify the system of Boterenbrood, D1, and Farsi by using the features, as taught by D2, in order to provide non-destructive bitwise arbitration on a CAN network,

where data is not lost during arbitration and provides an increased efficiency of the bus arbitration process (see D2 page 3, and 5).

Conclusion

12. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to KENAN CEHIC whose telephone number is (571)270-3120. The examiner can normally be reached on Monday through Friday 8:00-5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, KWANG BIN YAO can be reached on (571) 272-3182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Kenan Cehic/
Examiner, Art Unit 2616

/Kwang B. Yao/
Supervisory Patent Examiner, Art Unit 2616